Quality of Service Simulation

Quality of Service (QoS) is a means of ensuring high-quality performance for critical applications. The concept is that because requirements of some users and services are more critical than others, some traffic requires preferential treatment.

Using WAE Design QoS features, you can ensure that service levels are met without reactively expanding or over-provisioning the network. QoS features are available for undifferentiated traffic, for service classes alone, for interface queues alone, and for service classes that are mapped to queues.

- Undifferentiated traffic—Aggregate traffic on an interface.
- Service class—A user-defined classification of traffic that is not discovered by WAE. Examples include voice, video, and data. Service classes apply to the entire network unless you map them to specific queues.
- Queue—In live networks, traffic waits in conceptual lines (queues) and then is forwarded over an interface on a per-queue basis according to QoS parameters. Similarly in WAE Design, each queue has a set of user-defined QoS parameters (interface queue properties) that specify how these queues are prioritized and what percentage of traffic they carry. An interface contains zero or more queues that are discoverable by WAE. You can also manually create and configure them. The traffic per queue is also discovered.

This section contains the following topics:

QoS Parameters

In WAE Design, QoS requirements are defined by policies and interface queue properties (Figure 1: Policies and Interface Queue Parameters, on page 3).

Policy—Maximum percentage of traffic capacity that can be utilized by either a service class or by undifferentiated traffic. There are two policies: one for normal operation and one for worst-case scenarios. Policies set on service classes do not affect QoS requirements of any other service class. Nor would this parameter have any effect on live network behavior.

In addition to setting policies for the network, you can refine the policy to a group of interfaces (called an *interface policy group*). For example, you might need to model behavior for service classes offered only on interfaces of a specific capacity, or you might need to observe service class traffic on only one area of the network.

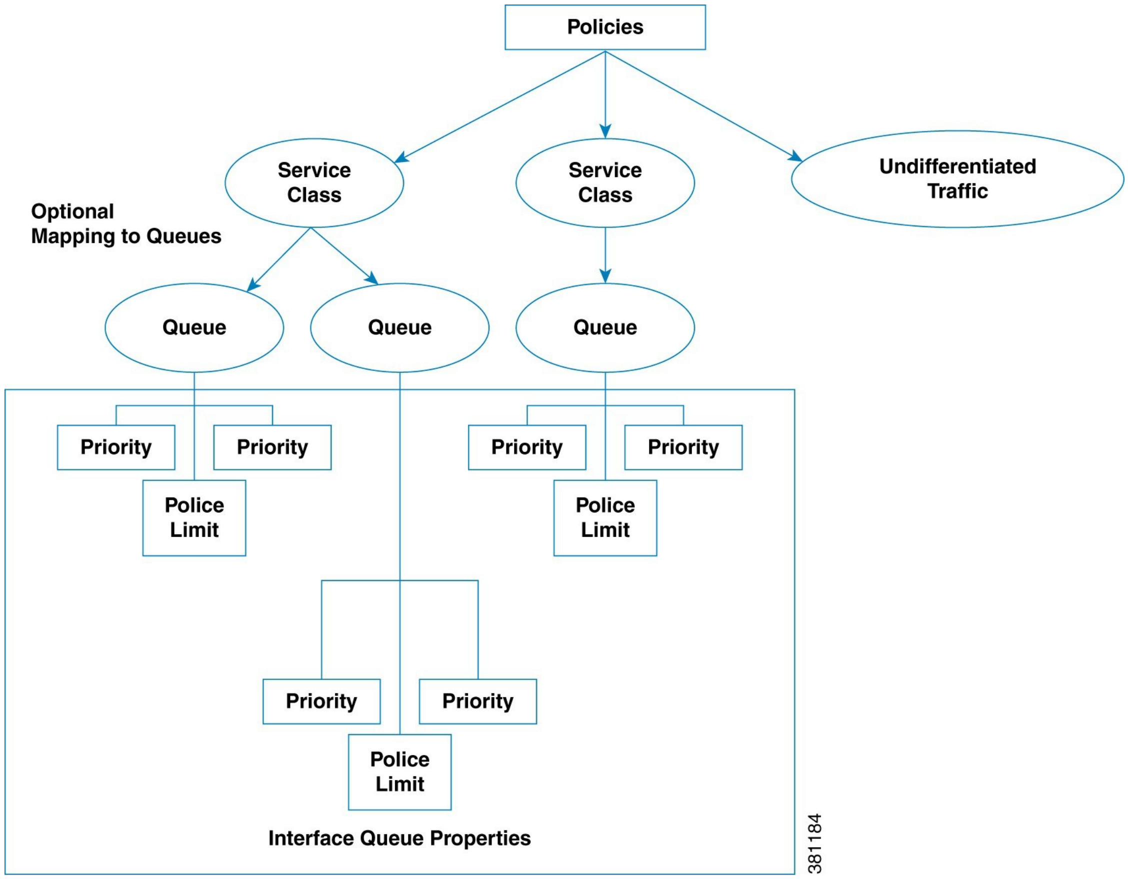
Interface queue properties—Configured parameters that would affect routing behavior in a live network. In WAE Design, the interface queue properties are priority, weight, and police limit.

- The *priority* identifies the precedence of the queue. For example, traffic in a priority 1 queue is routed before traffic in a priority 2 queue. Queues with the same priority evenly share the capacity based on weighted-round robin (WRR) calculations. You can change this behavior using the weight and police limit parameters. There are an unlimited number of priorities, though most networks only use no more than three. By default, queues do not have priorities.

- The *weight* is the percentage of preference given to queues of an equal priority level, which enables the network to fairly distribute the load among available resources. For example, if 10 Gbps were passing through a 10GbE interface on two priority 1 queues, by default 5 Gbps would pass through each queue. However, if you set the weight of one queue to 75% and the other to 25%, the distribution would be 7 Gbps and 2.5 Gbps, respectively. By default, all queues have a weight of 100%.

- The *police limit* is the maximum percentage of available capacity permitted through a queue of a given priority level, thereby preventing traffic from higher priority queues from starving lower priority queues. For example, if the interface is a 20GbE and a priority 1 queue has a police limit of 40%, then only 8 Gbps of interface traffic can go through this queue. By default, all queues have a police limit of 100%. To see examples of this *starvation* , refer to the examples in Policies and QoS Bound Calculations, on page 5, where you can see that lower priority queues received zero traffic due to priority settings.

Figure 1: Policies and Interface Queue Parameters



QoS Bound and QoS Violations

WAE Design uses the concepts of *QoS bound* and *QoS violation* as a way of identifying whether QoS parameters are being met or surpassed, thus better enabling you to plan for service requirements across the network. Policies and queue properties determine the QoS bound calculation. In turn, this calculation determines whether there is a violation.

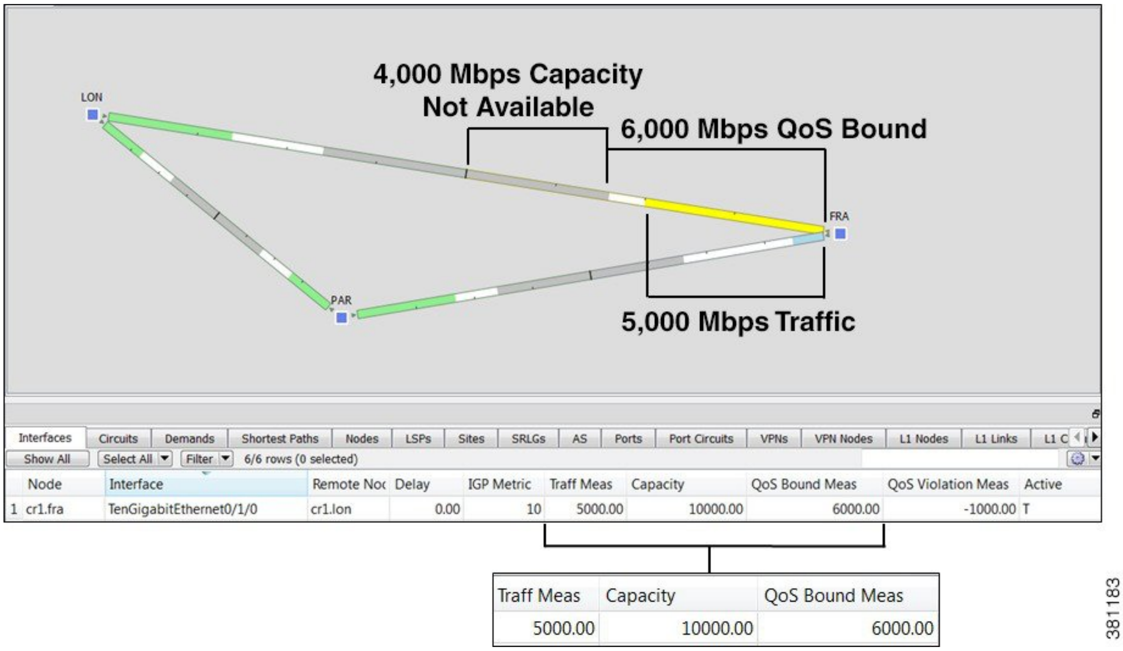
QoS Bound—Maximum interface capacity available without violating these QoS requirements. A separate QoS bound is calculated for both policy and interface queue properties.

QoS Bound for...	Calculation Based on...
Interface Queues	Combination of interface queue properties, or in live networks, it is the capacity percentage that is discovered.
Service class	Policy
Service class mapped to queues	The lower of these two calculations is used: • Policy for service class • Queue properties for queues

QoS Bound for...	Calculation Based on...
Undifferentiated traffic	Policy

In the plot, the QoS bound on an interface is a combination of the color and white (Figure 2: QoS Bound and Available Interface Capacity, on page 4). The interface capacity that is not available because it exceeds the QoS bound is in gray. Columns that convey the QoS bound information are QoS Bound Meas, QoS Bound Meas (%), QoS Bound Sim, and QoS Bound Sim (%).

Figure 2: QoS Bound and Available Interface Capacity

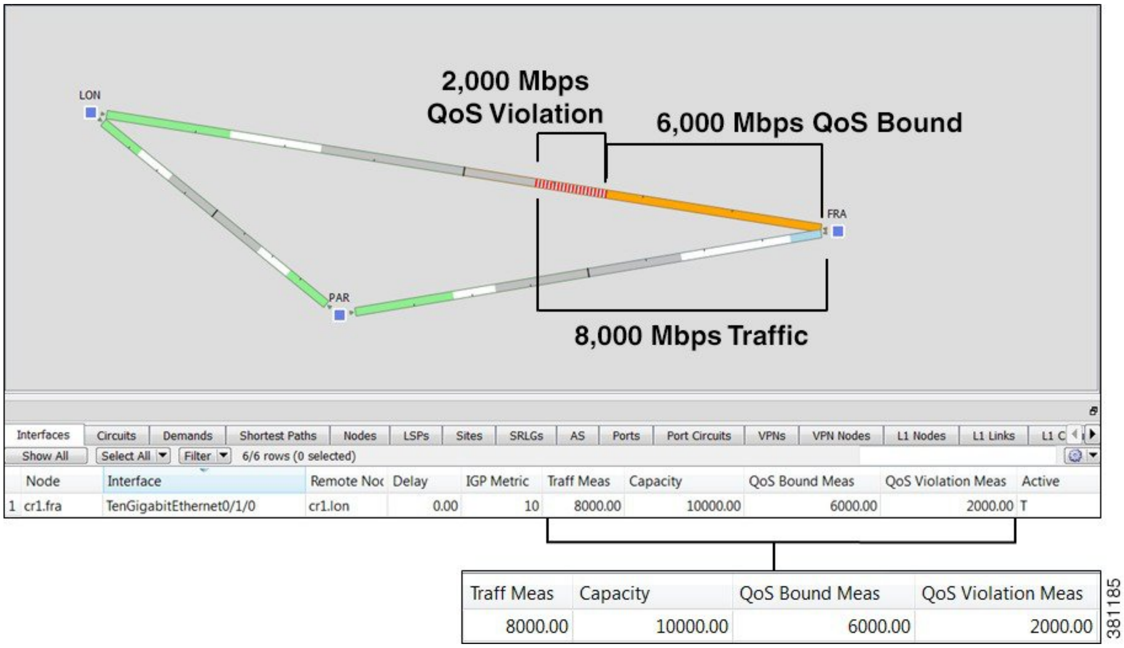


QoS bound calculations are a set of decisions being made to determine how to raise traffic on the queues until that traffic cannot be raised any further. This capacity, or the reason the traffic cannot be raised further, is defined both by the QoS parameters and the amount of traffic. For example, when traffic arrives at Queue X, WAE Design fixes the traffic on all other queues and then determines how it can raise the traffic on Queue X until some other traffic blocks it.

For those queues that do not reach full capacity, unused queue capacity is made available for other queues.

QoS Violation—Total traffic minus the capacity permitted for the queue (QoS bound). A violation occurs if the maximum QoS capacity allotted through policies and interface queue properties is exceeded. If the number appearing in the QoS Violation column is positive, the allotted capacity has been surpassed. If the number is negative, the allotted capacity has not been reached. In the plot, traffic exceeding the QoS bound appears in red and white stripes on the interface in violation (Figure 3: Example QoS Violation, on page 5).

Figure 3: Example QoS Violation



Policies and QoS Bound Calculations

If no other QoS parameters are set via the interface queue properties of priority, weight, and police limit, the QoS bound is equivalent to the policy set.

Table 1: Example Policies and QoS Bound Calculations

Example Configuration	QoS Bound	QoS Violation (Positive # = Violation)
Interface capacity = 10,000 Mbps Undifferentiated traffic = 5000 Mbps Normal operation policy = 60%	6000 Mbps (60%)	–1000 Mbps (–10%) Because this number is negative, there is no capacity violation (Figure 2: QoS Bound and Available Interface Capacity, on page 4).
Interface capacity = 10,000 Mbps Undifferentiated traffic = 8000 Mbps Normal operation policy = 60%	6000 Mbps (60%)	2000 Mbps (20%) Because this number is positive, there is a capacity violation (Figure 3: Example QoS Violation, on page 5).
Interface capacity = 10,000 Mbps Voice traffic = 6000 Mbps Video traffic = 2000 Mbps Voice normal operation policy = 90% Video normal operation policy = 60%	Voice = 9000 Mbps (90%) Video = 6000 Mbps (60%)	Voice = –3000 Mbps (30%) Video = –4000 Mbps (40%)

Interface Queue Properties and QoS Bound Calculations

WAE Design simultaneously calculates QoS bound for each queue in the interface. In so doing, WAE Design uses the interface queue parameters (priority, weight, and police limit) and the traffic measured or simulated for all queues in the interface. Priority is always considered first. If there are queues of equal priority, then weight is applied next.

- Queues with priority 1 share all available interface capacity. Their weight and police limits further refine how much each priority 1 queue can use (their QoS bound). Each priority 1 queue can borrow available capacity from other priority 1 queues up to the limit of their QoS bound.
- The available capacity for priority 2 queues is the total interface capacity less all capacity consumed by priority 1 queues. The process then begins again for all priority 2 queues. Their weight and police limits determine their QoS bound, and priority 2 queues can borrow capacity from each other up to the limits set by the QoS bound.
- This process continues for each successive priority level. Traffic that is outside any QoS bound is dropped to the lowest priority of all traffic on the interface.

For discovered networks with measured traffic, if no WAE Design QoS parameters are set, the QoS bound is based on whatever capacity percentages the live network has for each queue.

Priority

Provided policies are not set that further affect the QoS bound, a queue's QoS bound is calculated as follows:

- Priority 1 QoS bound = 100% of the interface capacity.
- Priority 2 QoS bound = Total interface capacity – amount of traffic consumed by priority 1 queues.
- Priority 3 QoS bound = Total interface capacity – amount of traffic consumed by (priority 1 + priority 2 queues).
- QoS bound for each succeeding priority follows this same pattern where the traffic consumed by all higher priority queues is subtracted from the total interface capacity.

Table 2: Examples of Priority QoS Bound Calculations

Example Configuration	QoS Bound	QoS Violation (Positive # = Violation)	QoS Bound Calculations
Interface capacity = 20,000 Mbps EF traffic = 6000 Mbps; priority = 1 BE traffic = 3000 Mbps; no priority set	EF = 20,000 Mbps BE = 14,000 Mbps	EF = –14,000 Mbps BE = –11,000 Mbps	EF = Total interface capacity because it is the only priority 1 queue BE = 20,000 (interface capacity) – 6000 (consumed by higher priority queues)
Interface capacity = 10,000 Mbps EF Traffic = 6000 Mbps; priority = 1 BE Traffic = 5000 Mbps; priority = 2	EF = 10,000 Mbps BE = 4000 Mbps	EF = –4000 Mbps BE = 1000 Mbps	EF = Total interface capacity because it is the only priority 1 queue BE = 10,000 (interface capacity) – 6000 (consumed by higher priority queues)

Weight

The weight identifies the forwarding precedence for queues of equal priority. If weights for queues of the same priority do not add up to 100%, weights are converted proportionally so they do add up to 100%.

Table 3: Examples of Weight QoS Bound Calculations

Example Configuration	QoS Bound	QoS Violation (Positive # = Violation)	QoS Bound Calculations
Interface capacity = 10,000 Mbps AF1 traffic = 3000 Mbps; priority = 1; weight = 100% AF2 traffic = 6000 Mbps; priority = 1; weight = 100%	AF1 = 5000 Mbps AF2 = 7000 Mbps	AF1 = –2000 Mbps AF2 = –1000 Mbps	AF1 = Half of capacity for priority 1 queues because both queues have equal weights AF2 = 5000 (half of capacity) + 2000 (unused AF1 capacity)
Interface capacity = 10,000 Mbps AF1 = 5000 Mbps; priority = 1; weight = 60% AF2 traffic = 6000 Mbps; priority = 1; weight = 40%	AF1 = 6000 Mbps AF2 = 5000 Mbps	AF1 = –1000 Mbps AF2 = 1000 Mbps	AF1 = 60% of capacity for all priority 1 queues AF2= 10,000 (interface capacity) – 5000 (consumed by AF1 queue)

Police Limits

Priority 1 queues have 100% of the interface traffic, and thus starve out the remaining queues. To prevent this queue starvation, use police limits to configure how much of the maximum percentage should be available for a given priority level.

Table 4: Examples of Police Limit QoS Bound Calculations

Example Configuration	QoS Bound	QoS Violation (Positive # = Violation)	QoS Bound Calculations
Interface capacity = 10,000 Mbps EF traffic = 1000 Mbps; priority = 1; police limit = 50% BE traffic = 2000 Mbps; priority = 2	EF = 5000 Mbps BE = 9000 Mbps	EF = –4000 Mbps BE = –7000 Mbps	EF = 50% of total interface capacity BE = 10,000 (interface capacity) – 1000 (capacity consumed by EF)
Interface capacity = 10,000 Mbps EF traffic = 1000 Mbps; priority = 1; police limit = 5% BE traffic = 2000 Mbps; priority = 2	F = 500 Mbps BE = 9500 Mbps	EF = 500 Mbps BE = –7500 Mbps	EF = 5% of total interface capacity BE = 10,000 (interface capacity) – 500 (capacity consumed by EF)

Example Configuration	QoS Bound	QoS Violation (Positive # = Violation)	QoS Bound Calculations
Interface capacity = 10,000 Mbps EF = 3000 Mbps; priority = 1; police limit = 20% AF1 traffic = 4000 Mbps; priority = 2; police limit = 75% AF2 traffic = 2500 Mbps; priority = 2; police limit 25%	EF = 2000 Mbps AF1 = 6000 Mbps AF2 = 4000 Mbps	EF = 1000 Mbps AF1 = –2000 Mbps AF2 = –1500 Mbps	EF = 20% of total interface capacity AF1 = 75% of (10,000 [interface capacity] – 2000 [capacity consumed by EF]) AF2 = 10,000 (interface capacity) – 2000 (capacity consumed by EF) – 4000 (capacity consumed by AF1)

Interface QoS Bound Calculations Using Multiple QoS Parameters

WAE Design calculates a QoS bound for interface queues based on all three parameters if they are all configured: priority, weight, and police limits.

Table 5: Examples of Interface QoS Bound Calculations Using Multiple QoS Parameters

Example Configuration	QoS Bound	QoS Violation (Positive #= Violation)	QoS Bound Calculation
Interface capacity = 10,000 Mbps EF = 3000 Mbps; priority = 1; police limit = 20% AF1 traffic = 4000 Mbps; priority = 2; weight = 75% AF2 traffic = 2500 Mbps; priority = 2; weight = 25%	EF = 2000 Mbps AF1 = 6000 Mbps AF2 = 4000 Mbps	EF = 1000 Mbps AF1 = –2000 Mbps AF2 = –1500 Mbps	EF = 20% of total interface capacity AF1 = Maximum of these two values. • 75% of (10,000 [interface capacity] – 2000 [capacity consumed by EF]) • 8000 (available capacity) – 2500 (AF2 traffic) AF2 = Maximum of these two values. • 25% of (10,000 [interface capacity] – 2000 [capacity consumed by EF]) • 8000 (available capacity) – 4000 (AF1 traffic)

Service Class QoS Bound Calculations Using Multiple QoS Parameters

If service classes have policies and they are mapped to queues, WAE Design calculates a QoS bound for both. WAE Design then uses the lowest value of the two so as to enforce restrictions in the strictest possible manner.

Example:

Interface capacity = 10,000 Mbps

QoS bound for service class = 50%, or 5000 Mbps based on policy

QoS bound for EF queue = 7500 Mbps based on combined parameters of priority, weight, and police limit

The QoS bound for this service class is 5000 Mbps because the policy QoS bound calculation is lower.

Viewing Queue and Service Class Information

Table 6: Queue and Service Class Information

To View	Show or Select
Queue information	Show the Interface Queue table. Select the queue from the QoS drop-down menu. Both the plot and the Traff Meas and Traff Sim columns display traffic data specific to the queue type selected.
Per-queue traffic in the Interfaces table	Select the queue from the QoS drop-down menu. Both the plot and the Traff Meas and Traff Sim columns display data specific to the queue type selected.
Per-service-class traffic	Select the service class from the QoS drop-down menu. Both the plot and the Traff Meas and Traff Sim columns in the Interfaces table display data specific to the service class selected.
Service class demands	Show the Service Class column in the Demands table.

Viewing QoS Bounds and QoS Violations

As Figure 2: QoS Bound and Available Interface Capacity, on page 4 and Figure 3: Example QoS Violation, on page 5 demonstrate, the QoS bounds and QoS violations appear in the plot view. Table 7: QoS Bounds and QoS Violations , on page 9 lists the available column options to display numeric values of the QoS bound calculations. For information on QoS values as they relate to VPNs, see VPN Simulation.

Table 7: QoS Bounds and QoS Violations

To View	Show This Column in Interfaces, Circuits, or Interface Queues Table
Measured Traffic	
Maximum capacity before a QoS bound is violated under normal operations	QoS Bound Meas
QoS bound as a percentage of total interface capacity	QoS Bound Meas (%)
QoS violations under normal operations; if the number is positive, there is a violation	QoS Violation Meas
QoS violation as a percent of the total interface capacity	QoS Violation Meas (%)
Simulated Traffic	
Maximum capacity before a QoS bound is violated under normal operations	QoS Bound Sim
QoS bound as a percentage of total interface capacity	QoS Bound Sim (%)

To View	Show This Column in Interfaces, Circuits, or Interface Queues Table
QoS violations under normal operations; if the number is positive, there is a violation	QoS Violation Sim
QoS violation as a percent of the total interface capacity	QoS Violation Sim (%)
Worst-Case Traffic	
Maximum capacity before a QoS bound is violated under worst-case operations	WC QoS Bound
WC QoS bound as a percentage of total interface capacity	WC QoS Bound (%)
QoS violations under worst-case operations; if the number is positive, there is a violation	WC QoS Violation
WC QoS violation as a percent of the total interface capacity	WC QoS Violation (%)
Service class causing the worst-case utilization	WC Service Class

Creating Service Classes

Step 1 Open the Manage QoS dialog box in one of two ways:

- Choose **Edit** > **Manage QoS**.
- Choose **Manage QoS** from the QoS drop-down menu in the toolbar.

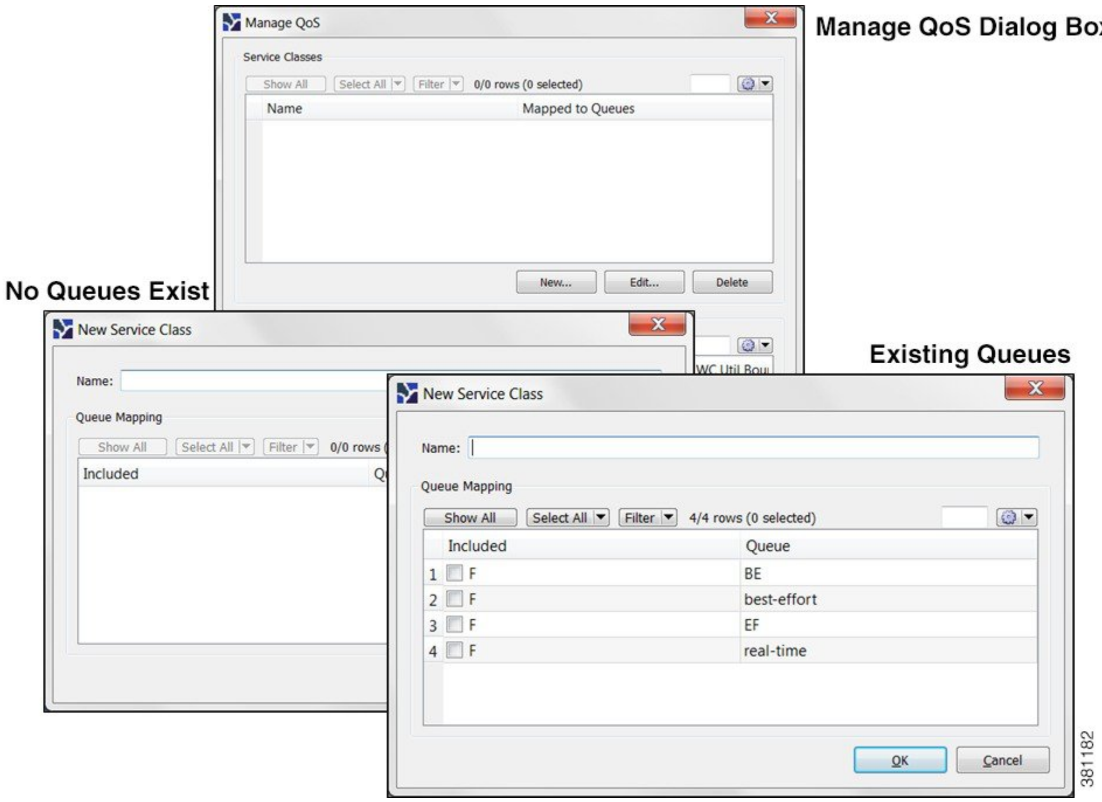
Step 2 Click **New**. The New Service Class dialog box that opens contains queues if they have already been discovered or if they have been manually created. Otherwise, the dialog box is empty. For instructions on how to create queues, see Creating Queues, on page 11.

a) Enter a unique name.
b) (Optional) If queues exist and if you want to map this new service class to one or more queues, select them from the list. If queues do not exist, but you want them, you must manually create the queues and then return to this dialog box to select them. See Mapping Service Classes to Queues, on page 12.
c) Click **OK**.

Step 3 Click **OK** to save and exit.

What to do next

Figure 4: Create Service Classes



Creating Queues

WAE Design discovers queues. However, you can manually add them. Once discovered or created, queues appear in the Interface Queues table.

Step 1 By default, queues are assigned to all interfaces. If you want this new queue to be assigned to specific interfaces, you must first select one or more interfaces.

Step 2 Open the New Interface Queues Properties dialog box in one of two ways:

- Choose **Insert** > **Interface Queues**.
- Right click in the plot area choose **New** > **Interface Queues**.

Step 3 Enter the queue name.

Step 4 (Optional) Enter the queue properties of priority, weight, and police limit. For information on how these queue properties behave, see Interface Queue Properties and QoS Bound Calculations, on page 6.

Step 5 Click **OK**. The new queue appears as an option in the QoS drop-down menu in the toolbar, as well as in the Manage QoS dialog box.

Step 6 (Optional) Map a service class to the queue. For instructions, see Mapping Service Classes to Queues, on page 12.

Another way to create queues is to edit the name of an existing one. For instructions, see Assigning Queues to Interfaces, on page 12.

Assigning Queues to Interfaces

The easiest way to assign queues to interfaces is to select the interface before creating the queue (see Creating Queues, on page 11). However, you can follow these steps to reassign the queue to different interfaces.

Step 1 From the Interface Queues table, select one or more queues.

Step 2 Double-click; a Properties dialog box opens.

Step 3 From the Node list, select the node associated with the interface.

Step 4 From the Interface list, select the interface to which you are assigning the queue.

Step 5 (Optional) Change the queue name or add a new one.

Step 6 (Optional) Specify QoS requirements in the Priority, Weight, and Police Limit fields. By default, queues do not have a priority and both their weight and police limits are 100%. For information on how these properties behave, see Interface Queue Properties and QoS Bound Calculations, on page 6.

Step 7 Click **OK**.

Mapping Service Classes to Queues

To map service classes to queues, those queues must first exist either because they were discovered by WAE or because you manually added them.

Step 1 Open the Manage QoS dialog box in one of two ways:

- Choose **Edit** > **Manage QoS**.
- Choose **Manage QoS** from the QoS drop-down menu in the toolbar.

Step 2 From the Service Classes list, select one service class.

Step 3 Click **Edit**.

Step 4 In the Edit Service Class dialog box, select one or more queues. Click **OK**.

Step 5 Repeat for each service class to which you are mapping queues.

Step 6 Click **OK** to save and exit.

Creating or Editing Policy Groups for Interfaces

Creating a policy group for interfaces lets you set policies for the group in the Manage QoS dialog box.

Step 1 Select one or more interfaces.

Step 2 Double-click to open the Properties dialog box.

Step 3 Click the **Advanced** tab.

Step 4 In the QoS Policy Group field, you have three options:

- To add this interface to an existing policy group, select it from the drop-down list.
- To change the name of an existing policy, select it and then type over it.
- To add a new policy group, select the empty row in the drop-down list and then type the name of the new group.

Step 5 Click **OK**.

Step 6 To assign a service class to this policy group, choose **Manage QoS** from the QoS drop-down menu in the toolbar.

Creating or Editing Service Class Policies

You can configure policies for undifferentiated traffic and for service classes.

Step 1 Open the Manage QoS dialog box in one of two ways:

- Choose **Edit** > **Manage QoS**.
- Choose **Manage QoS** from the QoS drop-down menu in the toolbar.

Step 2 In the Service Class Policies area, click **New**, or select a service class row and click **Edit**.

a) If creating a policy for undifferentiated traffic, select that option. If creating a policy for an existing service class, select it from the Service Class drop-down list.

b) (Optional) To apply this service class mapping to a group of interfaces, select from or enter the name in the Interface Policy Group drop-down list. You can enter a name that does not exist and then create the policy group for a set of interfaces. For further instructions, see Creating or Editing Policy Groups for Interfaces, on page 12.

c) In the Normal Operation field, enter the percentage of bandwidth capacity that you do not want this interface (or group of interfaces) to exceed for this traffic or service class under normal conditions.

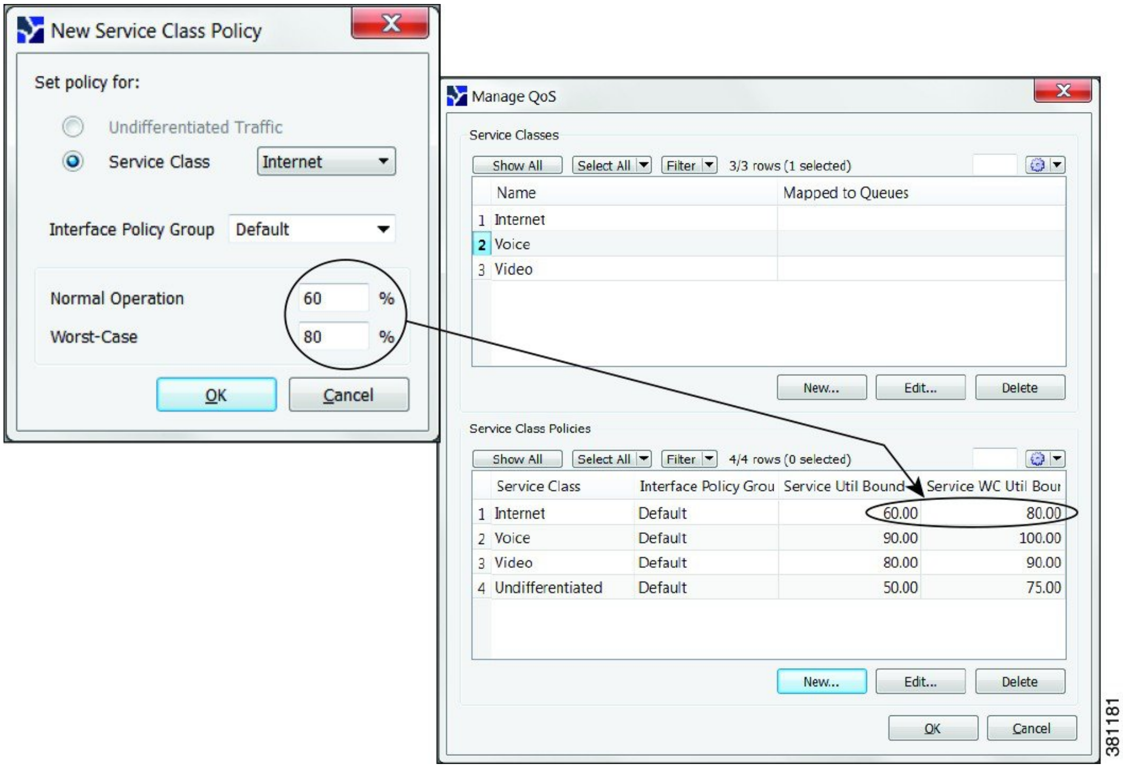
d) In the Worst-Case field, enter the percentage of bandwidth capacity that you do not want this interface (or group of interfaces) to exceed for this traffic or service under worst-case operating conditions.

e) Click **OK**. These values now appear in the Manage QoS dialog box where you can edit them as needed.

Step 3 Click **OK** to save and exit.

What to do next

Figure 5: Manage QoS Dialog Box



Editing Interface Queues Properties (QoS Requirements)

The three parameters are priority, weight, and police limit.

Step 1 In the Interface Queues table, right-click one or more queues and choose **Properties**.

Step 2 In the Properties dialog box, change one or more fields to create the desired QoS requirement.

Step 3 Click **OK**.